REMARKS

This application has been carefully reviewed in light of the final Office Action dated September 6, 2006. Claims 1 to 14 and 21 are in the application, of which claim 21 has been added, and claims 1, 9, 10 and 14 have been amended. Claims 1, 9 and 14 are the independent claims. Reconsideration and further examination are respectfully requested.

Initially, the Applicant notes that, since support for the substance of new claim 21 is found throughout the disclosure, including at least pages 3 and 8 of the specification, no new matter has been added herein.

Furthermore, the Applicant's undersigned representative thanks Examiner Dwivedi and Primary Examiner Wong for the thoughtful courtesies and kind treatment afforded during the personal interview conducted on November 1, 2006. In the interview, Applicant's representative discussed how the applied art is not seen to disclose the features recited by the independent claims, including, *inter alia*, the features that *i)* a constant-sized sorted result buffer is filled with a first N number of records from the data store, *ii)* remaining records in the data store are iteratively compared against a Nth record in the sorted result buffer, and *iii)* the Nth record in the sorted result buffer is iteratively replaced with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the Nth record in the sorted result buffer. As a result, Examiner Dwivedi graciously indicated that, if Applicants were to amend the independent claims to further clarify that the sorted result buffer is a uniform (or constant-sized) buffer, the rejections over the applied art would be overcome. Having amended the claims in accordance with the Examiner's suggestion, the Applicant respectfully request reconsideration and withdrawal of the outstanding rejection.

In the Office Action, the specification was objected to for allegedly failing to provide proper antecedent basis for the term "formatted." Without conceding the propriety or correctness of this objection, claim 10 has been amended herein to replace the term "formatted" with the term "formulated." Support for the feature of "formulating" a query using standard query language (SQL) is also found throughout the disclosure, including at least pages 2 and 11 of the

specification. Reconsideration and withdrawal of the objection to the specification are respectfully requested.

Claims 1 to 14 were rejected under 35 U.S.C. § 103(a) over M. Carey et al., "On Saying 'Enough Already!' In SQL," PROCEEDINGS OF THE 1997 ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA (Tucson, AZ)(Vol. 26, No. 2, June. 1997, p. 219-30)("Carey") in view of U.S. Patent No. 6,795,817 ("Agarwal"). As indicated above, independent claims 1, 9 and 14 have been amended herein as proposed by the Examiner. Accordingly, reconsideration and withdrawal of the § 103 rejection are respectfully requested.

The present disclosure generally relates to satisfying limit and order queries. A limit and order query is received that includes both of an order criteria and a limit criteria, the limit criteria specifying a maximum number N of records for a result set of records satisfying the limit and order query, and filling a constant-sized sorted result buffer with a first N number of records from a data store. The sorted result buffer is iteratively ordered based upon the order criteria, and remaining records in the data store are iteratively compared against a Nth record in the sorted result buffer based upon the order criteria. Additionally, the Nth record in the sorted result buffer is iteratively replaced with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the Nth record in the sorted result buffer, and the sorted result buffer is output as the result set of records.

Referring to particular claim language, independent claim 9 recites a method for satisfying limit and order queries. The method includes receiving a limit and order query that includes both of an order criteria and a limit criteria, the limit criteria specifying a maximum number N of records for a result set of records satisfying the limit and order query, and filling a constant-sized sorted result buffer with a first N number of records from a data store. The method further includes iteratively ordering the sorted result buffer based upon the order criteria, and iteratively comparing remaining records in the data store against a Nth record in the sorted result buffer based upon the order criteria. Moreover, the method includes iteratively replacing the Nth record in the sorted result buffer with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the Nth record in the sorted result buffer, and outputting the sorted result buffer as the result set of records.

The applied art is not seen to disclose, teach, or to suggest the foregoing features recited by the independent claims. In particular, neither Carey nor Agarwal, either alone or in any proposed combination (assuming *arguendo* that such a combination is possible) are seen to disclose at least the features that *i)* a constant-sized sorted result buffer is filled with a first N number of records from the data store, *ii)* remaining records in the data store are iteratively compared against a Nth record in the sorted result buffer, and *iii)* the Nth record in the sorted result buffer is iteratively replaced with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the Nth record in the sorted result buffer.

Carey describes a proposed SQL extension that would allow a query writer to limit the cardinality of a query result, such as a query which processes at most some number of result tuples. *See* Carey, *Abstract* and *Introduction*. As acknowledged by the Office Action, however, Carey is not seen to teach at least the features that *i)* a constant-sized sorted result buffer is filled with a first N number of records from the data store, *ii)* remaining records in the data store are iteratively compared against a Nth record in the sorted result buffer, and *iii)* the Nth record in the sorted result buffer is iteratively replaced with a remaining record in the data store based upon iteratively comparing remaining records in the data store against the Nth record in the sorted result buffer.

Agarwal is not seen to remedy the deficiencies of Carey. In particular, Agarwal discloses improving response times for a query which is executed against partitioned databases, where only a portion of each of the partitions are accessed during each pass through the partitions, and in which only retrieved portions of the partitions are processed. *See* Agarwal, col. 2, ll. 64 to col. 3, ll. 2; and *Abstract*. A specified number of rows are retrieved from each partition, where the number of rows to retrieve from each of P partitions equals the number of rows that must be returned to satisfy the query (or N). *See* Agarwal, col. 5, ll. 11 to 15 and 39 to 46. Accordingly, column 304, a type of sorting buffer, is seen to be filled with $P * N$ rows to satisfy a limit query for N records. *See* Agarwal, col. 6, ll. 17 to 19; and FIG. 3. Since the number of partitions P in Agarwal is always seen to be two or more, the sorted result buffer is not seen to be filled with the first N number of records, but rather the first $P * N$ number of records.

Once the first set of results are returned to the query, further rows of records are seen to be retrieved into the sorted result buffer. *See* Agarwal, col. 6, ll. 43 to 45. As illustrated in FIG. 6, during sorting, the size of the sorted result buffer begins with six records (column 304), then reduces in size to four records (column 308), and then increases in size to eight records (column 312). Since the sorted result buffer thus changes in size, it is not seen to be proper to describe the sorted result buffer of Agarwal as "constant-sized."

Finally, in Agarwal, records in the result buffer are either seen to remain in the buffer, or be eventually returned to the user. *See* Agarwal, col. 7, ll. 18 to 40. Consequently, Agarwal is not seen to describe the iterative comparison of remaining records in the data store against an Nth record, nor the iterative replacement of the Nth record in the sorted result buffer.

Accordingly, based on the foregoing amendments and remarks, independent claims 1, 9, 14 are believed to be allowable over the applied references. The other rejected claims in the application are each dependent from the independent claims and are believed to be allowable over the applied references for at least the same reasons. Because each dependent claim is deemed to define additional aspects of the disclosure, however, the individual consideration of each on its own merits is respectfully requested.
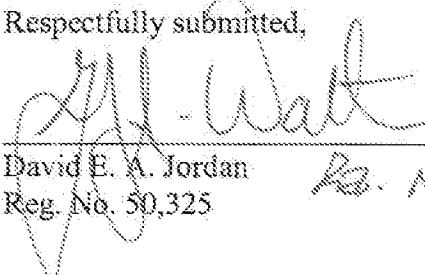
In view of the foregoing amendments and remarks, the entire application is believed to be in condition for allowance and such action is respectfully requested at the Examiner's earliest convenience.

No fees are believed to be due at this time. Please apply any other charges or credits to deposit account 06 1050.

Respectfully submitted,

Date: _December 6, 2006_

David E. A. Jordan
Reg. No. 50,325

Reg. No. 41,366

Fish & Richardson P.C.
1425 K Street, N.W.
11th Floor
Washington, DC 20005-3500
Telephone: (202) 783-5070
Facsimile: (202) 783-2331
40371245.doc